

CLAIMS

What is claimed:

1 1. A virtual storage system for linking a host to one or more
2 storage devices over a network, the system comprising:
3 an agent connected to the host, the agent having volatile memory
4 for storing a first copy of a table, the table having entries to map virtual
5 disk positions to locations on the storage devices; and
6 a controller coupled to the agent, the controller having non-volatile
7 memory for storing a second copy of the table, the controller intermittently
8 causing contents of the first copy of the table to be replaced by contents of
9 the second copy of the table,
10 whereby during an input/output (I/O) operation, the host accesses
11 one of the entries in the table stored on the agent to determine one of the
12 storage device locations.

1 2. The system of claim 1, wherein the table entries further
2 include an indication of whether an invalid state is activated such that the
3 invalid state for a table entry becomes activated when that table entry
4 contains no useable mapping information.

1 3. The system of claim 2, wherein the agent does not allow the
2 host to complete the I/O operations with one of the entries if the invalid
3 state for that entry is activated.

1 4. The system of claim 1, wherein the table entries further
2 include an indication of whether a no-write state is activated such that the
3 no-write state for one of the entries becomes activated when data cannot
4 be written to the storage location contained in that entry.

4 accessing a table mapping the block to a storage location on a
5 storage device;
6 issuing a corresponding operation to the storage device, wherein the
7 corresponding operation correlates to the operation on the virtual disk;
8 completing the corresponding operation; and
9 presenting the completed corresponding operation to the virtual
10 disk.

1 25. The method of claim 24, wherein the issuing step includes
2 issuing the corresponding operation from an agent coupled to the host.

1 26. The method of claim 24, further comprising updating the
2 table with a persistently-stored table residing in a non-volatile memory.

1 27. The method of claim 24, further comprising determining
2 states of the table.

1 28. The method of claim 24, further comprising sending a fault
2 message when the table is unable to be accessed.

1 29. The method of claim 24, further comprising storing the table
2 in a volatile memory.

1 30. The method of claim 24, receiving updates for the table from
2 a controller.

1 31. A method for maintaining a table for mapping virtual disk
2 blocks to storage locations on storage devices within a network,
3 comprising:
4 receiving a command from a controller at an agent storing the table;
5 activating states within entries of the table;
6 completing operations at the table; and
7 updating the table in response to the command.

1 32. The method of claim 31, further comprising setting a blocking
2 flag until operations are completed.

1 33. The method of claim 31, further comprising obtaining
2 mapping information from one of the entries in the table.

1 34. A computer program product comprising a computer useable
2 medium having computer readable code embodied therein for performing
3 an operation on a virtual disk coupled to a host within a network, the
4 computer program product adapted when run on a computer to effect steps
5 including:

6 specifying a block on the virtual disk within the operation;

7 accessing a table mapping the block to a storage location on a
8 storage device;

9 issuing a corresponding operation to the storage device, wherein the
10 corresponding operation correlates to the operation on the virtual disk;

11 completing the corresponding operation; and

12 presenting the completed corresponding operation to the virtual
13 disk.

1 35. A computer program product comprising a computer useable
2 medium having computer readable code embodied therein for maintaining
3 a table for mapping virtual disk blocks to storage locations on storage
4 devices within a network, the computer program product adapted when
5 run on a computer to effect steps including:

6 receiving a command from a controller at an agent storing the table;

7 activating states within entries of the table;

8 completing operations at the table; and

9 updating the table in response to the command.